IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Patent Application of	**Confirmation No.: 5531**
Shingo FUKUI	Date: November 6, 2008
Serial No.: 10/767,780	Group Art Unit: 2163
Filed: January 28, 2004	Examiner: Patrick A. DARNO
For: FILE SYSTEM FOR MANAGING FILES IN TREE STRUCTURE ALLOWING USERS TO READILY KNOW AVAILABILITY CONDITION	

VIA EFS-WEB
Mail Stop Appeal Brief Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF PURSUANT TO 37 C.F.R. §41.37

Sir:

This Brief is submitted in support of Applicants' appeal from the Examiner's rejection of this application dated May 5, 2008.

I. REAL PARTY IN INTEREST

The real party in interest in the above-identified application is: NEC Corporation, a corporation organized under the laws of Japan, the assignee of this application.

II. RELATED APPEALS AND INTERFERENCES

The applicants, the assignee, and the undersigned attorneys are not aware of any related appeals and interferences.

III.   **STATUS OF CLAIMS**

Claims 1-37 are pending and on appeal herein.

All pending claims have been previously presented. Claims 1, 12-18 and 20-37 have been previously amended. Claims 2-11 and 19 are original.

IV.   **STATUS OF AMENDMENTS**

All amendments have been entered. The fourth Office Action on the merits was issued on May 5, 2008. A Notice of Appeal was filed on July 31, 2008.

V.   **SUMMARY OF CLAIMED SUBJECT MATTER**

In an operating system such as Windows and Unix, information owned by users and/or groups is stored in a file system. The file system holds nodes of individual information linked in a tree structure. The node refers to each piece of information which forms part of a tree structure. In such conventional information sharing system, an owner can freely set an access to each node for an access permitted party. However, when a multiplicity of pieces of information are managed by the information sharing system, complicated and time-consuming processes are required to be performed in order to set an access permitted party for each of the nodes.

Broadly stated, the method and apparatus defined by Claims 1-37 is directed to overcome the above problems.

Independent claim 1, directed to the method according to the invention, recites:

Claim Feature	Support in Specification
An information sharing method for holding information owned by at least one unit user on a storage device in a tree structure provided for each unit user, said tree structure including a plurality of nodes sequentially arranged from a home root node to a plurality of leaf nodes, such that said information corresponds to each of said nodes to manage an availability condition of each of said nodes, said method comprising:	Page 20, line 24 - page 21, line 3; throughout pages 30 - 51

Claim Feature	Support in Specification
a first step in which a computer refers to the availability condition of each of said nodes on said storage device in response to an availability condition manipulation request for changing the availability condition of one of said nodes, to determine whether or not said availability condition manipulation request can be executed while satisfying a condition that while the availability condition can be changed at multiple nodes of the tree structure, the number of times of changes in the availability condition is limited to one at maximum along any of paths from said home root node to any one of the plurality of leaf nodes;	page 24, lines 6-14; page 34, line 1 - page 35, line 18; Fig. 25, steps S200-S210
a second step in which said computer executes the availability condition manipulation request such that said condition is satisfied when the availability condition manipulation request is determined as executable in said first step, and provides a determination that the availability condition manipulation request is not executable when the availability condition manipulation request is determined as not executable in said first step; and	page 35, line 22 - page 38, line 1; Fig. 26, steps S301-S306
a third step in which said computer refers to said availability condition in response to a tree structure manipulation request for modifying said tree structure, and executes the tree structure manipulation request such that said condition continues to be satisfied.	page 38, line 4 - page 39, line 26; Fig. 27, steps S401-S412

Independent claim 20, directed to the apparatus according to the invention, recites:

Claim Feature	Support in Specification
An information sharing apparatus for holding information owned by at least one unit user on a storage device in a tree structure provided for each unit user, said tree structure including a plurality of nodes sequentially arranged from a home root node to a plurality of leaf nodes, such that said information corresponds to each of said nodes to manage an availability condition of each of said nodes, said apparatus comprising:	Throughout pages 20-30; Figs. 12 - 19
an execution possibility determining module adapted, responsive to an availability condition manipulation request for changing the availability condition of one of said nodes, to refer to the availability condition of each of said nodes on said storage device to determine whether or not said availability condition manipulation request can be executed while satisfying a condition that while the availability condition can be changed at multiple nodes of the tree structure, the number of times of changes in the availability condition is limited to one at maximum along any of paths from said home root node to any one of the plurality of leaf nodes;	Fig. 19, element 11; page 28, lines 22-25
an availability condition manipulating module adapted to execute the availability condition manipulation request such that said condition is satisfied when said execution possibility determining module determines that the availability condition manipulation request is executable, and to provide a determination that the availability condition manipulation request is not executable when the availability condition manipulation request is determined as not executable by said execution possibility determining module; and	Fig. 19, element 13; page 29, lines 2-5

Claim Feature	Support in Specification
a tree structure manipulating module adapted, responsive to a tree structure manipulation request for modifying said tree structure, to refer to said availability condition to execute the tree structure manipulation request such that said condition continues to be satisfied.	Fig. 19, element 12; page 29, lines 6-10

Independent claim 37, directed to the apparatus according to the invention, recites:

Claim Feature	Support in Specification
A processor-readable medium incorporating a program of instructions configured to cause a computer to hold information owned by at least one unit user on a storage device in a tree structure provided for each unit user, said tree structure including a home root node, a plurality of leaf nodes, and a plurality of nodes sequentially arranged from the home root node to each leaf node, such that said information corresponds to each of said nodes to manage an availability condition of each of said nodes, said program of instructions comprising:	Throughout pages 20 - 51; Figs. 12 - 19, 25-27
first instructions configured to refer to the availability condition of each of said nodes on said storage device in response to an availability condition manipulation request for changing the availability condition of one of said nodes, to determine whether or not said availability condition manipulation request can be executed while satisfying a condition that while the availability condition can be changed at multiple nodes of the tree structure, the number of times of changes in the availability condition is limited to one at maximum along any selected path from said home root node to any one of the plurality of leaf nodes;	Fig. 19, element 11; Fig. 25, steps S200-S210; page 24, lines 6-14; page 28, lines 22-25; page 34, line 1 - page 35, line 18;

Claim Feature	Support in Specification
second instructions configured to execute the availability condition manipulation request such that said condition is satisfied when the availability condition manipulation request is determined as executable in said first processing, and to provide a determination that the availability condition manipulation request is not executable when the availability condition manipulation request is determined as not executable by the first instructions; and	Fig. 19, element 13; Fig. 26, steps S301-S306; page 29, lines 2-5; page 35, line 22 - page 38, line 1
third instructions configured to refer to said availability condition in response to a tree structure manipulation request for modifying said tree structure, and executing the tree structure manipulation request such that said condition continues to be satisfied.	Fig. 19, element 12; Fig. 27, steps S401-S412; page 29, lines 6-10; page 38, line 4 - page 39, line 26

VI.   GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

1. Whether Claims 1-8, 18, 20-27, 35 and 37 were properly rejected under 35 U.S.C. §102(b) as being anticipated by non-patent literature article titled "Improving the Granularity of Access Control for Windows 2000" authored by Michael M. Swift et al. (hereinafter "Swift").[1]

VII.   ARGUMENT

A.   Rejection of Claims 1, 20 and 37 under 35 U.S.C. §102(b)

Independent Claims 1, 20 and 30, as amended, recite the method, apparatus and a program of instructions, respectively, for managing availability of information stored in a tree structure "including a plurality of nodes sequentially arranged from a home root node to a plurality of leaf nodes." Further, each independent claim recites a condition requiring "that while the availability condition can be

---

1.   The issue of whether Claims 9-17 and 28-34 were properly rejected under 35 U.S.C. §103(a) as being unpatentable over Swift in view of U.S. Patent Publication No. 2004/0186845 or the issue of whether Claims 19 and 36 were properly rejected under 35 U.S.C. §103(a) as being unpatentable over Swift in view of U.S. Patent Publication No. 2003/0187854 are not separately addressed herein.

changed at multiple nodes of the tree structure, the number of times of changes in the availability condition is limited to one at maximum along any selected path from said home root node to any one of the plurality of leaf nodes." These limitations of Claims 1, 20 and 37 are not disclosed in the cited prior art.

Swift discloses the mechanisms in Windows 2000 that enable fine-grained and centrally managed access control for operating system components and applications. An example of the hierarchy of information discussed in Swift is shown in Fig. 5. Here, the "Company" container represents the home root node, and the "Jane User" object represents a leaf node. Each object and each container has a set of properties. Swift teaches that there are properties that are common to many types of objects and that, instead of giving access right to a particular object, users may be given access to particular properties of an object. This is particularly illustrated in the Example on pages 14-15. Specifically, the first ACE grants administrators full control over this user object, the second ACE grants group administrators read and write access to user's public information, and the third ACE grants a user herself an access to change the password. Accordingly, access to a particular set of properties is controlled by this set's administrator. In other words, each node of Fig. 5 may have several administrators depending on the properties of this node.

The Examiner believes that Swift discloses the limitation of Claims 1, 20 and 37 requiring that along a path from the home root node to the leaf node the availability condition changes not more than once. Applicants respectfully disagree.

Swift states:

> Type-specific inheritance and static inheritance allow centralized management by propagating changes through a hierarchy of objects, so that access control changes are made **in one place**. These features support delegation by allowing an administrator to grant access to **a single type of object**, or even a single property on a single type of object. In addition, that access is propagated both to existing objects and to new objects when they are created. Swift, page 25, third paragraph.

In other words, with reference to Fig. 5 of Swift, access control changes to each of the individual "Jane User" objects is "made in one place." However, this does not mean that in any path of the hierarchy from the home root node "Company" to any of the leaf nodes "Jane User", the number of

times of changes in the availability condition is limited to one at maximum. Swift specifically indicates that these access control changes are different for different nodes. Specifically, as Swift disclosed in reference to its Fig. 5, the "Company" node has properties that will only be accessible by the company administrators, the "Departments" node has properties that will only be accessible by company administrators and department administrators, the "Research" node has properties that will only be accessible by company administrators, department administrators and group administrators, and, finally, the Jane User node has properties that will only be accessible by company administrators, department administrators, group administrators and Jane User. These multiple changes in access availability are possible because, as Swift repeatedly indicates, "despite presenting data as a hierarchy, the Active Directory internally stores data in a flat database and maintains indexes" over the names and properties of the objects. Therefore, while the changes are made from "one place" in the Archive Directory, the availability condition along a particular path may change multiple times in the system of Swift.

Contrary to the apparatus and method of Swift, Claims 1, 20 and 37 define a true hierarchical tree structure and recite that the availability condition can be changed at **multiple nodes** of the tree structure but, along any selected path from the home root node to any one of the leaf nodes, the availability condition may only change once.

In view of the above, Applicant respectfully submits that the limitation of Claims 1, 20 and 37 requiring that the availability condition can be changed at multiple nodes of the tree structure and the number of times the availability condition changes on any path from a home root node to any leaf node is limited to one at maximum, is not disclosed or suggested by Swift. Accordingly, as Swift fails to anticipate each and every limitation of Claims 1, 20 and 37, it is respectfully requested that the rejection of Claims 1, 20 and 37 and their dependent Claims 2-19 and 21-36 under 35 U.S.C. §103(a) be withdrawn.

VIII. CONCLUSION

Claims 1, 20 and 37 and their dependent claims should be deemed allowable over the prior art of record.

Credit card payment for the required filing fee in the amount of $510.00 (large entity) is submitted via EFS-WEB.

If this communication is filed after a time period has elapsed and no separate Petition is enclosed, the Commissioner of Patents and Trademarks is petitioned, under 37 C.F.R. §1.136(a), to extend the time for filing this Brief by the number of months which will avoid abandonment under 37 C.F.R. §1.135. The fee under 37 C.F.R. §1.17 should be charged to our Deposit Account No. 15-0700.

In the event the actual fee is greater than the payment submitted or is inadvertently not enclosed or if any additional fee during the prosecution of this application is not paid, the Patent Office is authorized to charge the underpayment to Deposit Account No. 15-0700.

Respectfully submitted,

THIS CORRESPONDENCE IS BEING
SUBMITTED ELECTRONICALLY
THROUGH THE UNITED STATES
PATENT AND TRADEMARK OFFICE
EFS FILING SYSTEM
ON NOVEMBER 6, 2008

MAX MOSKOWITZ
Registration No.: 30,576
OSTROLENK, FABER, GERB & SOFFEN, LLP
1180 Avenue of the Americas
New York, New York 10036-8403
Telephone: (212) 382-0700

<u>**CLAIMS APPENDIX**</u>

1.   **(Previously Presented)**   An information sharing method for holding information owned by at least one unit user on a storage device in a tree structure provided for each unit user, said tree structure including a plurality of nodes sequentially arranged from a home root node to a plurality of leaf nodes, such that said information corresponds to each of said nodes to manage an availability condition of each of said nodes, said method comprising:

a first step in which a computer refers to the availability condition of each of said nodes on said storage device in response to an availability condition manipulation request for changing the availability condition of one of said nodes, to determine whether or not said availability condition manipulation request can be executed while satisfying a condition that while the availability condition can be changed at multiple nodes of the tree structure, the number of times of changes in the availability condition is limited to one at maximum along any of paths from said home root node to any one of the plurality of leaf nodes;

a second step in which said computer executes the availability condition manipulation request such that said condition is satisfied when the availability condition manipulation request is determined as executable in said first step, and provides a determination that the availability condition manipulation request is not executable when the availability condition manipulation request is determined as not executable in said first step; and

a third step in which said computer refers to said availability condition in response to a tree structure manipulation request for modifying said tree structure, and executes the tree structure manipulation request such that said condition continues to be satisfied.

2.   **(Original)** The information sharing method according to claim 1, wherein said first step includes:

when said availability condition manipulation request involves setting an availability condition, determining that said availability condition manipulation request is executable when the availability condition of a node under manipulation is the same as that of the home root node, or is a change start point of the availability condition in said tree structure, and determining that said

availability condition manipulation request is not executable when the availability condition of said node under manipulation is different from that of said home root node, and is not said change start point.

3. **(Original)** The information sharing method according to claim 1, wherein said first step includes:

when said availability condition manipulation request involves clearing an availability condition, determining that said availability condition manipulation request is executable when a node under manipulation is a change start point of the availability condition in said tree structure, and determining that said availability condition manipulation request is not executable when said node under manipulation is not said change start point.

4. **(Original)** The information sharing method according to claim 1, wherein said first step includes:

determining that said availability condition manipulation request is not executable when a node under manipulation intended by said availability condition manipulation request is a home root node.

5. **(Original)** The information sharing method according to claim 1, wherein said second step includes:

when said availability condition manipulation request involves setting an availability condition, setting the availability condition of a node under manipulation as requested by said availability condition manipulation request, and setting the same availability condition to all nodes included in a maximum partial tree in which said node under manipulation is in position of a root.

6. **(Original)** The information sharing method according to claim 1, wherein said second step includes:

when said availability condition manipulation request involves clearing availability condition, clearing the availability of a node under manipulation, and setting the same availability

condition as that of said node under manipulation to all nodes included in a maximum partial tree in which said node under manipulation is in position of a root.

7. **(Original)** The information sharing method according to claim 1, wherein said third step includes:

when said tree structure manipulation request involves creating a new node, creating said new node at a requested location.

8. **(Original)** The information sharing method according to claim 7, wherein said third step further includes:

setting the same availability condition of a parent node of said new node to said new node after creating said new node.

9. **(Original)** The information sharing method according to claim 1, wherein said third step includes:

when said tree structure manipulation request involves duplicating a node group comprising at least one node, creating a duplicate of said node group at a requested location.

10. **(Original)** The information sharing method according to claim 9, wherein said third step further includes:

setting the same availability condition set to the parent node of a root node of said node group to said nodes which make up the duplicate of said node group after creating the duplicate of said node group.

11. **(Original)** The information sharing method according to claim 1, wherein said third step includes:

when said tree structure manipulation request involves moving a node group comprising at least one node, moving said node group to a location under a requested destination node.

12.   **(Previously presented)**   The information sharing method according to claim 11, wherein said third step further includes:

performing one processing operation of a plurality of different availability condition modification processing operations depending on the availability condition of each of said nodes included in said node group after moving said node group.

13.   **(Previously presented)**   The information sharing method according to claim 12, wherein said plurality of different availability condition modification processing operations comprises processing for maintaining the availability condition of each of said nodes included in said node group, processing for setting the same availability condition of said destination node to each of said nodes, and processing for querying a user whether said processing for maintaining the availability condition or said processing for setting the same availability condition is performed.

14.   **(Previously presented)**   The information sharing method according to claim 11, wherein said third step further includes:

performing one processing operation of a plurality of different availability condition modification processing operations depending on whether the availability condition of said destination node is different from that of the home root node after moving said node group.

15.   **(Previously presented)**   The information sharing method according to claim 14, wherein said plurality of different availability condition modification processing operations comprises processing for maintaining the availability condition of each of said nodes included in said node group, processing for setting the same availability condition of said destination node to each of said nodes, and processing for querying a user whether said processing for maintaining the availability condition or said processing for setting the same availability condition is performed.

16.   **(Previously presented)**   The information sharing method according to claim 14, wherein said third step further includes:

performing one processing operation of a plurality of different availability condition modification processing operations depending on the availability condition of each of said nodes included in said node group after moving said node group.

17.   **(Previously presented)**   The information sharing method according to claim 16, wherein said plurality of different availability condition modification processing operations comprises processing for maintaining the availability condition of each of said nodes included in said node group, processing for setting the same availability condition of said destination node to each of said nodes, and processing for querying a user whether said processing for maintaining the availability condition or said processing for setting the same availability condition is performed.

18.   **(Previously presented)**   The information sharing method according to claim 1, wherein each of said nodes in said tree structure is classified into at least one of an unchanged node having the same availability condition as the home root node; a change start node having an availability condition different from that of said home root node and different from that of a parent node; and a change takeover node having an availability condition different from that of said home root node and the same as that of a parent node, said classification being added to information on said availability condition as a change state type of each of said nodes for management,

wherein said computer refers to said change state type for examining said availability condition.

19.   **(Original)**   The information sharing method according to claim 1, wherein said tree structure includes a node which is a short-cut to another node.

20.   **(Previously Presented)**   An information sharing apparatus for holding information owned by at least one unit user on a storage device in a tree structure provided for each unit user, said tree structure including a plurality of nodes sequentially arranged from a home root node to a plurality of leaf nodes, such that said information corresponds to each of said nodes to manage an availability condition of each of said nodes, said apparatus comprising:

an execution possibility determining module adapted, responsive to an availability condition manipulation request for changing the availability condition of one of said nodes, to refer to the availability condition of each of said nodes on said storage device to determine whether or not said availability condition manipulation request can be executed while satisfying a condition that while the availability condition can be changed at multiple nodes of the tree structure, the number of times of changes in the availability condition is limited to one at maximum along any of paths from said home root node to any one of the plurality of leaf nodes;

an availability condition manipulating module adapted to execute the availability condition manipulation request such that said condition is satisfied when said execution possibility determining module determines that the availability condition manipulation request is executable, and to provide a determination that the availability condition manipulation request is not executable when the availability condition manipulation request is determined as not executable by said execution possibility determining module; and

a tree structure manipulating module adapted, responsive to a tree structure manipulation request for modifying said tree structure, to refer to said availability condition to execute the tree structure manipulation request such that said condition continues to be satisfied.

21.   **(Previously presented)**   The information sharing apparatus according to claim 20, wherein said execution possibility determining module is operative when said availability condition manipulation request involves setting an availability condition to determine that said availability condition manipulation request is executable when the availability condition of a node under manipulation is the same as that of the home root node, or is a change start point of the availability condition in said tree structure, and to determine that said availability condition manipulation request is not executable when the availability condition of said node under manipulation is different from that of said home root node, and is not said change start point.

22.   **(Previously presented)**   The information sharing apparatus according to claim 20, wherein said execution possibility determining module is operative when said availability condition manipulation request involves clearing an availability condition to determine that said availability condition manipulation request is executable when a node under manipulation is a

change start point of the availability condition in said tree structure, and to determine that said availability condition manipulation request is not executable when said node under manipulation is not said change start point.

23. **(Previously presented)** The information sharing apparatus according to claim 20, wherein said execution possibility determining module determines that said availability condition manipulation request is not executable when a node under manipulation intended by said availability condition manipulation request is a home root node.

24. **(Previously presented)** The information sharing apparatus according to claim 20, further comprising an availability condition setting supporting module when called from said availability condition manipulating module for setting the same availability condition of a node under manipulation to all nodes included in a maximum partial tree in which said node under manipulation is in position of a root,

wherein said availability condition manipulating module is operative when said availability condition manipulation request involves setting an availability condition to set the availability condition of a node under manipulation as requested by said availability condition manipulation request, and to call said availability condition setting supporting module.

25. **(Previously presented)** The information sharing apparatus according to claim 20, further comprising an availability condition clear supporting module when called from said availability condition manipulating module for setting the same availability condition of a node under manipulation to all nodes included in a maximum partial tree in which said node under manipulation is in position of a root,

wherein said availability condition manipulating module is operative when said availability condition manipulation request involves clearing availability condition to clear the availability of a node under manipulation, and to call said availability condition clear supporting module.

26. **(Previously presented)** The information sharing apparatus according to claim 20, wherein said tree structure manipulating module is operative when said tree structure manipulation request involves creating a new node to create said new node at a requested location.

27. **(Previously presented)** The information sharing apparatus according to claim 26, further comprising a new node creation supporting module when called from said tree structure manipulating module for setting the same availability condition of a parent node to said new node,

wherein said tree structure manipulating module calls said new node creation supporting module after creating said new node.

28. **(Previously presented)** The information sharing apparatus according to claim 20, wherein said tree structure manipulating module is operative when said tree structure manipulation request involves duplicating a node group comprising at least one node to create a duplicate of said node group at a requested location.

29. **(Previously presented)** The information sharing apparatus according to claim 28, further comprising a duplication supporting module when called from said tree structure manipulating module for setting the same availability condition set to the parent node of a root node of said node group to said nodes which make up the duplicate of said node group,

wherein said tree structure manipulating module calls said duplication supporting module after creating the duplicate of said node group.

30. **(Previously presented)** The information sharing apparatus according to claim 20, wherein said tree structure manipulating module is operative when said tree structure manipulation request involves moving a node group comprising at least one node to move said node group to a location under a requested destination node.

31. **(Previously presented)** The information sharing apparatus according to claim 30, further comprising a movement supporting module when called from said tree structure

manipulating module for performing one processing operation of a plurality of different availability condition modification processing operations depending on the availability condition of each of said nodes included in said node group,

wherein said tree structure manipulating module calls said movement supporting module after moving said node group.

32. **(Previously presented)** The information sharing apparatus according to claim 31, wherein said plurality of different availability condition modification processing operations performed by said movement supporting module comprises processing for maintaining the availability condition of each of said nodes included in said node group, processing for setting the same availability condition of said destination node to each of said nodes, and processing for querying a user whether said processing for maintaining the availability condition or said processing for setting the same availability condition is performed.

33. **(Previously presented)** The information sharing apparatus according to claim 31, wherein said movement supporting module further performs one processing operation of a plurality of different availability condition modification processing operations depending on whether the availability condition of said destination node is different from that of the home root node.

34. **(Previously presented)** The information sharing apparatus according to claim 33, wherein said plurality of different availability condition modification processing operations performed by said movement supporting module comprises processing for maintaining the availability condition of each of said nodes included in said node group, processing for setting the same availability condition of said destination node to each of said nodes, and processing for querying a user whether said processing for maintaining the availability condition or said processing for setting the same availability condition is performed.

35. **(Previously presented)** The information sharing apparatus according to claim 20, wherein each of said nodes in said tree structure is classified into at least one of an

unchanged node having the same availability condition as the home root node, a change start node having an availability condition different from that of said home root node and different from that of a parent node; and a change takeover node having an availability condition different from that of said home root node and the same as that of a parent node, said classification being added to information on said availability condition as a change state type of each of said nodes for management,

wherein said information sharing apparatus refers to said change state type for examining said availability condition.

36. **(Previously presented)** The information sharing apparatus according to claim 20, further comprising a short-cut managing module for creating a node which is a short-cut to a referenced node, said short-cut managing module being responsive to designation of said short-cut node for searching said referenced node.

37. **(Previously Presented)** A processor-readable medium incorporating a program of instructions configured to cause a computer to hold information owned by at least one unit user on a storage device in a tree structure provided for each unit user, said tree structure including a home root node, a plurality of leaf nodes, and a plurality of nodes sequentially arranged from the home root node to each leaf node, such that said information corresponds to each of said nodes to manage an availability condition of each of said nodes, said program of instructions comprising:

first instructions configured to refer to the availability condition of each of said nodes on said storage device in response to an availability condition manipulation request for changing the availability condition of one of said nodes, to determine whether or not said availability condition manipulation request can be executed while satisfying a condition that while the availability condition can be changed at multiple nodes of the tree structure, the number of times of changes in the availability condition is limited to one at maximum along any selected path from said home root node to any one of the plurality of leaf nodes;

second instructions configured to execute the availability condition manipulation request such that said condition is satisfied when the availability condition manipulation request is

determined as executable in said first processing, and to provide a determination that the availability condition manipulation request is not executable when the availability condition manipulation request is determined as not executable by the first instructions; and

third instructions configured to refer to said availability condition in response to a tree structure manipulation request for modifying said tree structure, and executing the tree structure manipulation request such that said condition continues to be satisfied.

EVIDENCE APPENDIX

None.

RELATED PROCEEDINGS APPENDIX

None.